Upstream Relocation of the CCS

Upstream relocation moves the CCS to an NDC site that is closer to the client, such as the client workstation **42**, than the present CCS. A DTP response to a request to access data includes a "use ticket" that accompanies data which is being passed upstream from NDC site to NDC site. The DTP use ticket may be marked as USE_ONCE or USE_MANY depending upon whether the image of the data may remain cached at an NDC site after it has been used to respond to the request that caused the data to be fetched from downstream. The DTP use ticket for an image of data is always marked as USE_MANY when it begins its journey from the NDC server terminator site to the client site. However, as the image of the data passes upstream from NDC site to NDC site, its use may be restricted to USE_ONCE at any NDC site through which it passes. Thus, when the image of the data passes through the current CCS for the file **156** the channel **116** at that NDC site changes the data's DTP use ticket from USE_MANY to USE_ONCE.

As the image of the file **156** is projected through successive NDC sites, if the DTP use ticket is marked as USE_MANY, the image of the data may remained cached within the NDC buffers **129** assigned to the channel **116** through which the image traverses the NDC site. Whether or not any data remains cached within the NDC buffers **129** assigned to the channel **116** after passing through the NDC site is determined solely by the local site. Maintaining a projected image of data at an NDC site is a resource allocation issue, and each NDC site must maintain control of its own resources. However, if the DTP use ticket is marked USE_ONCE, none of the data may remain cached within the NDC buffers **129** assigned to the channel **116** after traversing the NDC site.

Upstream relocation of the CCS due to a decease notification requires only that the current CCS recognize if it no longer has multiple upstream NDC sites engaged in CWS activities. When that occurs, the NDC site that formerly was the CCS merely stops marking the DTP use ticket USE_ONCE. This change in the marking of the DTP use ticket immediately permits upstream NDC sites to begin caching any images of the file **156** that may be projected into them in the future.

However, if one of the upstream NDC sites currently has additional upstream NDC sites that are creating a CWS condition, that NDC site will declare itself to be the new CCS and begin changing the DTP use ticket from USE_MANY to USE_ONCE. In this way, the NDC **50** of the present invention facilitates relocating the CCS upstream.

Downstream Relocation of the CCS

Relocating the CCS downstream moves the CCS to an NDC site closer to the NDC server terminator site **22**. Referring to FIG. **15**, if no clients are accessing the file **156** and then if a client on LAN **44B** requests access for writing the file **156** residing on the NDC server terminator site **22**, a projected image of the file **156** flows from NDC site **22**, through NDC sites **26A**, **202**, **204A**, and into NDC site **206**. The client may now read and write the projection of the file **156** present in the NDC client terminator site **206** with an unlimited number of simultaneous processes without the NDC client terminator site **206** checking with any of the downstream NDC sites **204A**, **202** or **26A**, or with the NDC server terminator site **22** before each operation. The NDC client terminator site **206** need communicate with the downstream NDC sites **204A**, **202**, **26A** and **22** only to load or unload data from the channel **116** at the NDC client terminator site **206**.

If a client on LAN 44A connected to the NDC site **204B** begins to access the file **156** for writing it, the NDC client

terminator site **204B** claims a channel **116** that then sends an NDC_LOAD message to intermediate NDC site **202**. The NDC_LOAD message from the channel **116** will indicate that NDC site **204B** is loading data that will be overlaid by a write operation. Upon processing this NDC_LOAD message, the NDC site **202** finds that a channel **116** already exists for the file **156**. The existing channel **116** identifies NDC site **204A** as a current upstream NDC site, and also indicates that the channel **116** for the file **156** is currently enabled. This combination of conditions implies that the CCS for the file **156**, if one exists, is located either at NDC site **204A** or at an NDC site upstream from NDC site **204A**. As described above, the upstream site structures **182** at the NDC site **202** not only identify all upstream NDC sites accessing the file **156**, they also indicate the type of file operations that have occurred at each NDC site accessing the file **156**. These few facts, i.e. the existence of a CWS condition and that the CCS is not currently located downstream from the NDC site **202** enable site **202** to determine that it should declare itself the ccs.

While holding off the write request from the NDC site **204B**, NDC site **202** recalls or disables all upstream NDC sites that are caching projected images of the file **156**. As described above, "disable" is sufficient for any NDC sites at which the file **156** was only being read. However, if there are any sites that have modified their image of the file **156**, their dirty data must flushed back to the new CCS, NDC site **202**. Therefore, NDC site **202** sends a recall message to NDC site **204A**.

Before NDC site **204A** responds to the recall message from NDC site **202**, NDC site **204A** transmits its own recall message upstream to NDC client terminator site **206**. After all of upstream NDC sites have responded to the recall message from NDC site **204A**, NDC site **204A** will respond back to NDC site **202**, forwarding any dirty data that had been soiled by NDC site **204A**, or by NDC sites upstream from NDC site **204A**.

After NDC site **204A** responds to the recall message from NDC site **202**, NDC site **202** can begin processing the write request from NDC site **204B**. NDC site **202** has now declared itself to be the CCS for file **156**. NDC site **202** is now in charge of sequencing all read/write operations that are requested for the file **156** by its own clients, and by clients of all upstream NDC sites, e.g. NDC sites **204A**, **204B** and **206**.

While the intermediate NDC site **202** remains the CCS with multiple connections to upstream NDC sites **204A** and **204B** at least one of which is writing the file **156**, no file data or metadata will be cached upstream of the intermediate NDC site **202**. If, after all NDC sites that were accessing the file **156** for writing have disconnected from the file **156**, the intermediate NDC site **202** as CCS still has one or more upstream NDC sites that are reading the file **156**, the CCS will relocate upstream as described above.

Industrial Applicability

Within a networked digital computer system, file servers, workstations, gateways, bridges, and routers are all potential candidates to become an NDC site. The NDC **50** is a software module that can easily be ported to different environments. The NDC **50** requires a minimum of 250 k bytes of RAM, of which 50 k is code and the remainder is allocated for various data structures and buffers. Each channel **116** occupies approximately 500 bytes of RAM. Thus, one megabyte of RAM can accommodate about two thousand channels **116**. At current memory prices, this amount of RAM costs well under $50. As illustrated in FIG. **4**, the structure for the subchannel **118** included in each channel

116 provides pointers to 18 NDC buffers **129**. In the preferred embodiment of the invention, each NDC buffer **129** stores 8 k bytes of projected data. Thus, the eighteen NDC buffers **129** associated with each channel **116** can store an image of up to 18*8 k bytes, i.e. 144 k bytes. Thus, with no additional subchannels **152**, each channel **116** can accommodate the complete projection, both of data and of NDC metadata, of any dataset of up to 144 k bytes in length.

An NDC site having only 250 k bytes RAM would be useful for only certain limited applications. Each site usually allocates anywhere from 4 to 256 megabytes of RAM for its NDC **50**. For example, a 128 megabyte NDC site that allocates 32 megabytes of RAM for NDC data structures can maintain over 50,000 simultaneous connections to data conduits **62** while also storing 96 megabytes of data image projections. Because accessing large datasets may require more than one channel **116**, the number of simultaneous dataset connections will vary depending on the mix of datasets which are currently being accessed.

With so many channels **116** packed into a single NDC site, the task of quickly connecting a new request to the channel **116** for the specified dataset, or claiming the least recently used channel **116** if there is none, might seem to be a daunting feat. However, the NDC **50** provides two mechanisms that facilitate solving this problem. The channel hash lists and the channel free list are methods of stringing together the channels **116** in such a way that any particular channel **116**, or the least recently used channel **116**, can be quickly located. Moreover, preferably the number of hash buckets allocated at each NDC site is adjusted so that, on the average, there are 4 channels **116** in each hash bucket. Limiting the number of channels **116** in each hash bucket to 4 permits quickly determining whether or not an NDC site presently has a channel **116** assigned to accessing a particular dataset.

If the NDC client terminator site **24** receives a request from the client workstation **42** to access a dataset for which the NDC client terminator site **24** is also the NDC server terminator site, and if the request seeks to access data that is not currently being projected into the NDC buffers **129** of the NDC site **24**, the delay in responding to the first request as measured at the client intercept routine **102** is approximately 25 milliseconds (about the same as for NFS). However, once the NDC **50** dispatches a response to the client workstation **42**, the site will employ intelligent, efficient, and aggressive read ahead to ensure that as long as the client workstation **42** continues to access the file sequentially, data will almost always be projected into the NDC buffers **129** of the NDC client terminator site **24** before the client workstation **42** requests to access it. By prefetching data in this manner, responses to most subsequent requests from the client workstation **42** can be dispatched from the NDC client terminator site **24** to the client workstation **42** within 100 microseconds from the time the NDC site **24** receives the request.

If the client workstation **42** requests to access a dataset that is at an NDC site other than the NDC client terminator site **24**, such as NDC sites **26**B, **26**A or **22**, responding to the first request from the client workstation **42** requires an additional 25 millisecond delay for each NDC site that must respond to the request. However, because the NDC client terminator site **24** attempts to pre-fetch data for the client workstation **42**, the NDC site **24** will dispatch responses to subsequent requests from the client workstation **42** in about 100 microseconds as described above.

While the presently preferred embodiment of the NDC **50** is implemented in software, it may also be implemented in firmware by storing the routines of the NDC **50** in a Read only Memory ("ROM"). Furthermore, the operation of the NDC **50** is independent of any particular communication hardware and protocol used to implement the LAN **44**, and of the filesystem that is used for accessing the hard disks **32**, **34** and **36**. Analogously, the operation of the NDC **50** is independent of the communication hardware and communication protocol by which DTP messages **52** pass between pairs of NDC sites **22**–**26**A, **26**A–**26**B, or **26**B–**24**. The communication hardware and protocols for exchanging DTP messages **52** include backplane buses such as the VME bus, local area networks such as Ethernet, and all forms of telecommunication. Accordingly, DTP messages **52** exchanged between NDC sites may pass through gateways, including satellite data links, routers and bridges.

While the NDC **50** has been described thus far in the context of a distributed multi-processor computer system **20** in which various NDC sites, such as the sites **22**, **26**A, **26**B and **24**, are envisioned as being separated some distance from each other, the NDC **50** may also be applied effectively within a single computer system that incorporates a network of computers. FIG. **16** depicts a file server referred to by the general reference character **300**. Those elements depicted in FIG. **16** that are common to the digital computer system **20** depicted in FIG. **1** carry the same reference numeral distinguished by a double prime ("''") designation. The file server **300** includes a host processor **302** for supervising its overall operation. Within the file server **300**, an internal bus **304**, perhaps a VME bus, couples the main host processor **302** to a pair of storage processors **306**A and **306**B. The storage processors **306**A–B control the operation of a plurality of hard disks **32**A" through **32**F". The internal bus **304** also couples a pair of file processors **312**A and **312**B, a pair of shared primary memories **314**A and **314**B, and a plurality of Ethernet processors **316**A through **316**D to the host processor **302**, to the storage processors **306**A–B, and to each other.

During the normal operation of the file server **300** without the incorporation of any NDCs **50**, the Ethernet processors **316**A–D receive requests to access data stored on the disks **32**A" through **32**F" from clients such as the client workstation **42** that is illustrated in FIG. 1. The requests received by the Ethernet processors **316**A–D are transferred to one of the file processors **312**A–B. Upon receiving a request to access data, the file processor **312**A or **312**B communicates with one of the storage processors **306**A or **306**B via the internal bus **304** to effect the transfer an image of the data from the disks **32**A" through **32**F" into the primary memories **314**A–B. After an image of the requested data has been transferred into the primary memories **314**A–B, the Ethernet processor **316** that received the request then transmits the requested data to the client thereby responding to the request.

The file processors **312**A–B may incorporate a hard disk cache located in the primary memories **314**A–B. The presence of a hard disk cache in the file server **300** allows it to respond to some requests to access data without any communication between one of the file processors **312**A–B and one of the storage processors **306**A–B. However, even though the file server **300** includes a hard disk cache, during operation of the file server **300** responding to each request to access data received by the Ethernet processors **316**A–D necessarily involves communications between the Ethernet processors **316**A–D and the file processors **312**A–B. That is, even though data needed by the Ethernet processors **316**A–D for responding to requests is already physically present in the primary memories **314**A–B, to gain access to the data the Ethernet processors **316**A–D must first communicate with the file processors **312**A–B because the data is stored in a hard disk cache under the control of the file processors **312**A–B.

To enhance the overall performance of the file server **300**, each of the Ethernet processors **316**A–D may incorporate an NDC **50** operating as NDC client terminator site. Each NDCs **50** included in the Ethernet processors **316**A–D accesses a set of NDC buffers **129** allocated within the primary memories **314**A–B. In addition to the NDCs **50** included in the Ethernet processors **316**A–D, the file server **300** may also include other NDCs **50** operating as NDC server terminator sites in the file processors **312**A–B. The NDCs **50** in the file processors **312**A–B also access a set of NDC buffers **129** allocated within the primary memories **314**A–B.

In a file server **300** so incorporating NDCs **50**, if one of the Ethernet processors **316**A–D receives a request to access data that is already present in the NDC buffers **129** of its NDC **50**, its NDC **50** may respond immediately to the request without communicating with an NDC **50** located in one of the file processors **312**A–B. Analogously, if one of the Ethernet processors **316**A–D receives a request to access data that is not present in its NDC buffers **129** but that is present in the NDC buffers **129** of the NDCs **50** in the file processors **312**A–B, those NDCs **50** may also respond immediately to the request without accessing the hard disk cache controlled by the file processors **312**A–B. Under such circumstances, the NDC **50** operating in the file processors **312**A–B may immediately respond to a request from the NDC **50** operating in the Ethernet processors **316**A–D merely by providing it with a pointer to the location of the data within the primary memories **314**A–B. Thus, by employing NDCs **50** both in the Ethernet processors **316**A–D and in the file processors **312**A–B, data that is physically present in NDC buffers **129** located in the primary memories **314**A–B becomes available more quickly to the Ethernet processors **316**A–D for responding to requests from clients such as the client workstation **42** by eliminating any need to access the hard disk cache controlled by the file processors **312**A–B.

Although the present invention has been described in terms of the presently preferred embodiment, it is to be understood that such disclosure is purely illustrative and is not to be interpreted as limiting. Consequently, without departing from the spirit and scope of the invention, various alterations, modifications, and/or alternative applications of the invention will, no doubt, be suggested to those skilled in the art after having read the preceding disclosure. Accordingly, it is intended that the following claims be interpreted as encompassing all alterations, modifications, or alternative applications as fall within the true spirit and scope of the invention.

What is claimed is:

1. In a network of digital computers that includes a plurality of Network Distributed Cache ("NDC") sites, each NDC site including an NDC that has an NDC buffer, a method for projecting an image of a stored dataset from an NDC server terminator site into an NDC client terminator site in response to a request to access such dataset transmitted from a client site to the NDC client terminator site, the method comprising the steps of:

(a) the NDC receiving the request to access data in the stored dataset;

(b) the NDC checking the NDC buffer at this NDC site to determine if a projected image of data requested from the dataset is already present there;

(c) if the NDC buffer of this NDC site does not contain a projected image of all data requested from the dataset, and if the NDC site receiving the request is not the NDC server terminator site for the dataset, the NDC of this NDC site transmitting a request for data from this NDC site downstream to another NDC site closer to the NDC server terminator site for the dataset than the present NDC site;

(d) if the NDC buffer of this NDC site does not contain a projected image of all data requested from the dataset, and if the NDC site receiving the request is the NDC server terminator site for the dataset, the NDC of this NDC site accessing the stored dataset to project an image of the requested data into its NDC buffer;

(e) repeating the steps (a) through (d) until the NDC buffer of the downstream NDC site receiving the request contains a projected image of all requested data;

(f) each successive NDC site, having obtained a projected image of all the requested data, returning data requested from it upstream to the NDC site from which it received the request until the requested data arrives at the NDC client terminator site; and

(g) the NDC client terminator site, upon receiving the requested data, returning the requested data to the client site.

2. The method of claim **1** wherein, prior to returning the requested data to the client site, the NDC client terminator site reformats the requested data from the protocol employed for communications between pairs of NDC sites into the protocol in which the client site requested access to the dataset from the NDC client terminator site.

3. The method of claim **2** wherein the NDC client terminator site selectively reformats the requested data from the protocol employed for communications between pairs of NDC sites into a particular one of a plurality of different protocols that matches the protocol in which the client site requested access to the dataset from the NDC client terminator site.

4. The method of claim **1** further comprising the steps of:

(h) each NDC upon receiving the initial request to access the dataset claiming a channel and designating the channel for storing various data relevant to processing requests to access the dataset;

(i) storing in the channel of this NDC site data collected by the NDC while processing requests to access the dataset; and

(j) if the NDC site is not performing any steps of the method that are required for responding to a request to access a dataset, the method at NDC sites further including:

i. periodically analyzing data stored in the channel of this NDC site to determine whether it is possible to anticipate future requests to access the dataset;

ii. if the analysis of data stored in the channel of this NDC site establishes that it is possible to anticipate future requests to access the dataset, further analyzing data stored in the channel to determine whether anticipated future requests will soon present this NDC site with another request to access the dataset and whether the projected image of data present in the NDC buffer includes sufficient data to respond immediately to the anticipated request; and

iii. if anticipated future requests to access the dataset will soon present this NDC site with another request to access the dataset and if the projected image of data present in the NDC buffer lacks sufficient data to respond immediately to the anticipated request, this NDC site, before receiving a request therefor, requesting data from the dataset from the next downstream NDC site.

5. The method of claim **4** wherein the NDC site, in requesting data from the next downstream NDC site, requests a quantity of data that is larger than the amount of data returned by the NDC client terminator site to the requesting client site in response to a request to access the dataset received by the NDC client terminator site from the client site.

6. The method of claim **1** wherein each NDC upon receiving the initial request to access the dataset claims a channel and designates the channel for storing various data relevant to processing requests to access the dataset, the method at NDC sites further comprising the step of:

(h) if the NDC is not performing any steps of the method that are required for responding to a request to access a dataset, the method at NDC sites further including:

i. periodically analyzing channels to determine if such channel is presently no longer needed for responding to requests to access the dataset for which the channel was claimed; and

ii. if the channel being analyzed is no longer needed for responding to requests to access the dataset, then processing the no longer needed channel to prepare it for immediate claiming in response to a subsequent request from a client site to access another dataset.

7. The method of claim **6** wherein the periodic analysis of channels to determine if such channel is presently no longer needed for responding to requests is halted if the number of channels available to respond to a future request to access a dataset exceeds a pre-established upper threshold.

8. The method of claim **7** wherein the NDC having stopped periodically analyzing channels, resumes periodically analyzing channels if the number of channels available to respond to a future request to access a dataset drops below a pre-established lower threshold that is less than the pre-established upper threshold.

9. A network of digital computers that includes a client site which requests access to a dataset that is stored at a location that can be accessed through the network, the network comprising:

a plurality of NDC sites, the dataset whose access is requested by the client site being stored at an NDC server terminator site, a request from the client site for access to the dataset being received by an NDC client terminator site, each NDC site including:

(a) an NDC that has an NDC buffer;

(b) means for the NDC to receive the request to access the dataset;

(c) means for the NDC to check the NDC buffer at this NDC site to determine if a projected image of data requested from the dataset is already present there wherein:

i. if the NDC buffer of this NDC site does not contain a projected image of all data requested from the dataset, and if this NDC site is not the NDC server terminator site for the dataset, the NDC including means for transmitting a request for data from this NDC site downstream to another NDC site closer to the NDC server terminator site for the dataset than the present NDC site;

ii. if the NDC buffer of this NDC site does not contain a projected image of all data requested from the dataset, and if this NDC site is the NDC server terminator site for the dataset, the NDC including means for accessing the dataset to project an image of the requested data into its NDC buffer; and

iii. if the NDC buffer of an NDC site contains a projected image of all requested data, the NDC including means for returning data requested from it upstream to the NDC site from which it received the request, whereby through a succession of such returns of data from one NDC site to the next upstream NDC site the requested data ultimately arrives at the NDC client terminator site; and

(d) data return means for returning the requested data from the NDC client terminator site to the client site.

10. The network of claim **9** wherein, prior to returning the requested data to the client site, said data return means reformats the requested data from the protocol employed for communications between pairs of NDC sites into the protocol in which the client site requested access to the dataset from the NDC client terminator site.

11. The network of claim **10** wherein the data return means selectively reformats the requested data from the protocol employed for communications between pairs of NDC sites into a particular one of a plurality of different protocols that matches the protocol in which the client site requested access to the dataset from the NDC client terminator site.

12. The network of claim **9** wherein each NDC site further comprises:

(e) means for the NDC, upon receiving the initial request to access the dataset, to claim a channel for storing various data relevant to processing requests to access the dataset;

(f) means for the NDC to store in the channel data collected by the NDC while processing requests to access the dataset;

(g) if the NDC site is not responding to a request to access dataset, means for the NDC to periodically analyze data stored in the channel to determine whether it is possible to anticipate future requests to access the dataset;

(h) if the analysis of data stored in the channel by the NDC establishes that it is possible to anticipate future requests to access the dataset, means for the NDC to further analyze data stored in the channel to determine whether anticipated future requests will soon present this NDC with another request to access the dataset and whether the projected image of data present in the NDC buffer includes sufficient data to respond immediately to the anticipated request; and

(i) if anticipated future requests will soon present this NDC site with another request to access the dataset and if the projected image of data present in the NDC buffer lacks sufficient data to respond immediately to the anticipated request, means for the NDC to request from the next downstream NDC site data from the dataset before receiving a request therefor.

13. The network of claim **12** wherein the NDC site requests from the next downstream NDC site a quantity of data from the dataset before receiving a request therefor, the quantity of data requested by the NDC site being larger than the amount of data returned by the NDC client terminator site to the requesting client site in response to a request to access the dataset received by the NDC client terminator site from the client site.

14. The network of claim **9** wherein each NDC site further comprises:

(e) means for the NDC, upon receiving the initial request to access the dataset, to claim a channel for storing various data relevant to processing requests to access the dataset;

(f) if the NDC is not responding to a request to access a dataset, means for the NDC to periodically analyze channels to determine if such channel is presently no longer needed for responding to requests to access the dataset for which the channel was claimed; and

(g) if analysis of the channel establishes that the channel is no longer needed for responding to requests to access the dataset, means for the NDC to process the no longer needed channel to prepare it for immediate claiming in response to a subsequent request from a client site to access another dataset.

15. The network of claim **14** wherein each NDC site further comprises means for halting the analysis of channels to determine if such channels are presently no longer needed for responding to requests if the number of channels available to respond to future requests to access datasets exceeds a pre-established upper threshold.

16. The network of claim **15** wherein each NDC site further comprises means for resuming the previously halted analysis of channels to determine if such channels are presently no longer needed for responding to requests if the number of channels available to respond to future requests to access datasets drops below a pre-established lower threshold that is less than the pre-established upper threshold.

* * * * *